Solution Brief

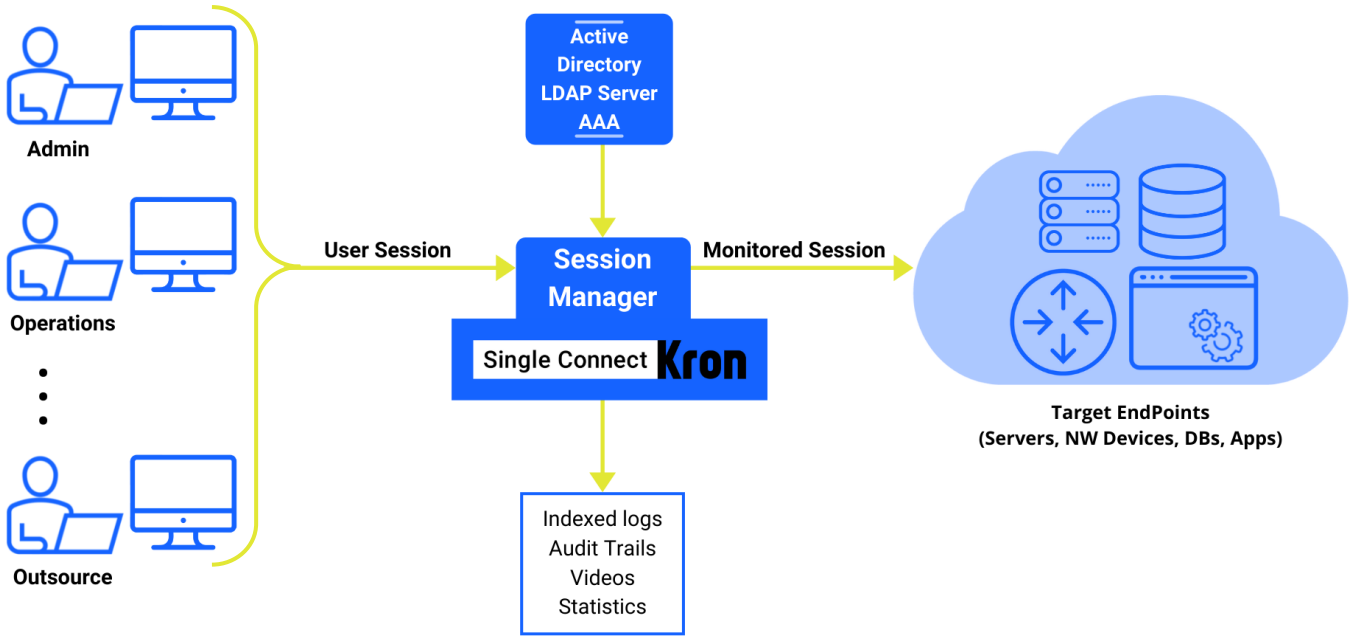
Single Connect Session Manager

Single Connect Kron

Introduction/Background

There are thousands of servers/devices and thousands of users (employees, contractors, etc.) who connect to them every day. It makes tens of thousands of connections between users and servers/devices every day, which is very complex and probably unmanageable from the security perspective. Not every connection (between user and device/servers) is at the same level of importance. For example, there are users (employees) you trust, and they connect to servers/devices that do not include critical enterprise assets. For such connection types, logging of which user connected to which device/server is may be sufficient. However, some connection types are very critical. For example, there are 3rd party technical support guys who connect to the most critical network/IT resources, and in such cases, you want to make sure that you have "full visibility" and "full control" on such connections. And session manager is the solution for this case.

Session Manager High-Level Topology



As shown in the figure above, Session Manager stands transparently in the middle of sessions and does not require any agent to be installed on the user PCs or target servers/applications. Session Manager supports the following command line interfaces: (SSH, Telnet), Remote Desktop Connections (RDP/VNC), Web sessions (HTTP/S), File transfer (SFTP), Database connections (SQL).

How Session Manager Works

Sample Flow for a CLI Connection

Step 1: User initiates a CLI session towards Single Connect with his/her own username and password.

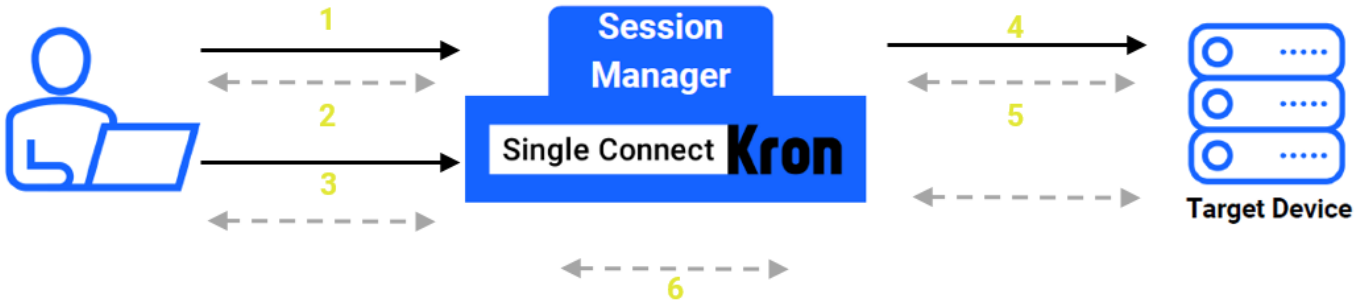
Step 2: A CLI session between User and Single Connect is established. Single Connect displays the list of devices that the user has the right to access.

Step 3: User selects the target device from the list that s/he wants to connect.

Step 4: This time, Single Connect initiates a CLI session towards the target device with a username/password.

Step 5: A CLI session between Single Connect and target device is established.

Step 6: Two separate CLI sessions (user<->Single Connect and Single Connect<->Target device) are back-to-back connected by the Session Manager. Now, Single connect is man-in-the-middle for the entire session duration. Everything goes through Single Connect; therefore, Single Connect has “full control” of the session and provides “full visibility.” When the user enters a command on his/her CLI screen, Single Connect receives it, processes it, and decides whether to forward the command to the target device or reject it.



Features and Benefits

- Full visibility. Detailed audit logs. All commands either failed or succeeded are logged. You know which user attempted to run which command on which device and when indisputably.
- “Separation of duties” and “least privilege” practices are achieved, regardless of the role/profile capabilities of the target device. Any custom policies (allowed command sets, blocked command sets) can be defined and applied to any user group to ensure that users can execute only the “required set of commands” to fulfill their tasks and no other command. It prevents standard users from having over-privileged access.
- Context-aware policy. For example, do not allow the “delete” command to run at the device level (higher/outer level at the command tree) but allow it to run at the port level (lower/inner level at the command tree)
- 2-factor authorization. You can enable command approval to require approval from a second person to run the command (e.g., shutdown command). So, when users enter the “shutdown” command, their supervisor receives an email, and if the supervisor clicks the approve link, the command is executed. Otherwise, it is rejected.
- Eliminates weak passwords and/or nonexpired passwords.
- Enables to define time-based access limitations. Based on the time of day, day of the week, maintenance-window hours, etc.
- Disables inactive privileged accounts.
- Session recording and video playback for forensic analysis
- Dual control (referred to as “four eyes” or “second eye”). When a user connects to a device, a supervisor can monitor the session in real-time and take/release the control of the session. This is good when real-time monitoring is required for emergency accounts or monitoring someone in the training phase.
- Single sign-on. Users connect to Single Connect with their username/password. And then select any -allowed- device to connect. Users don’t need to use/know separate usernames/passwords to connect to different devices/servers.
- Helps to eliminate password sharing and shared account usage. Users always log in with their own username/password even if the username/password used to connect to the device is a shared account. For example, a user connects to Single Connect with username=john and then selects a device to connect. Single Connect establishes a session towards the target device but maybe using username=admin. Users never see or know the real username/password used to connect to the target device; all they know is their own username/password.
- Ensures who is the real user connecting to the target device, indisputably.
- Auto-lock user account when an employee terminates employment. (Integration with enterprise Active Directory (or LDAP) is required.)